<u>IN THE CLAIMS</u>

Please amend claims 31, 39, 41, 49, 51, and 59 as indicated below.

1. – 30. (Cancelled).

31. (Currently Amended)   A method, comprising:

    receiving an instruction for updating a file having a plurality of fragments stored in a

        memory, the instruction including changed data for updating a first fragment of

        the file;

    creating a second fragment based on the first fragment and the changed data, the

        <u>second fragment representing an updated version of the first fragment</u>;

    copying a first table associated with the first fragment to form a second table, the

        second table having a handle referencing the second fragment; ~~and~~

    deleting the first fragment when the creating and copying operations are completed<u>;</u>

        <u>and</u>

    <u>processing further accesses to the deleted first fragment using the second fragment via</u>

        <u>the second table</u>.

32. (Previously presented)   The method of claim 31, further comprising marking, via the

first table, the first fragment as invalid while the creating and copying operations are being

performed.

33. (Previously presented)   The method of claim 32, further comprising:

determining whether the instruction has been cancelled before the creating and

copying operations are completed;

marking the first fragment as valid if the instruction has been cancelled before the

creating and copying operations are completed; and

deleting the second fragment and the second table after marking the first fragment as

valid.

34. (Previously presented)   The method of claim 31, further comprising deleting the first

table when the creating and copying operations are completed.

35. (Previously presented)   The method of claim 31, further comprising marking, via the

second table, the second fragment as a backup copy of the first segment while the creating

and copying operations are being performed.

36. (Previously presented)   The method of claim 35, further comprising unmarking the

second fragment as the backup copy of the first segment after the creating and copying

operations are completed.

37. (Previously presented)   The method of claim 31, wherein the copying the first table

comprises writing the handle into a replacement entry.

38. (Previously presented)   The method of claim 37, wherein the writing the handle

comprises marking a handle pointing to the first fragment as invalid and marking the

handle pointing to the second fragment as valid.

39. (Currently Amended)   ~~The method of claim 31, wherein the first table is a member of a~~ ~~first chain having a plurality of tables, the method further comprises:~~

A method, comprising:

receiving an instruction for updating a file having a plurality of fragments stored in a

memory, the instruction including changed data for updating a first fragment of

the file;

creating a second fragment based on the first fragment and the changed data;

copying a first table associated with the first fragment to form a second table having a

handle referencing the second fragment, the first table being a member of a

first chain having a plurality of tables;

deleting the first fragment when the creating and copying operations are completed;

creating a copy for each of the plurality tables in the first chain to form a second chain,

each of the copied table having a handle pointing to the next copied table in the

second chain;

replacing the first chain with the second chain; and

deleting the first chain.

40. (Previously presented)   The method of claim 39, wherein each table in the first and

second chains is hierarchically ranked at a different level and wherein each table is capable

of associating with one or more tables in the respective chain via one or more handles

respectively.

41. (Currently Amended)   A machine-readable medium having executable code to cause a

machine to perform a method, the method comprising:

   receiving an instruction for updating a file having a plurality of fragments stored in a

      memory, the instruction including changed data for updating a first fragment of

      the file;

   creating a second fragment based on the first fragment and the changed data, the

      second fragment representing an updated version of the first fragment;

   copying a first table associated with the first fragment to form a second table, the

      second table having a handle referencing the second fragment; and

   deleting the first fragment when the creating and copying operations are completed;

      and

   processing further accesses to the deleted first fragment using the second fragment via

      the second table.

42. (Previously presented)   The machine-readable medium of claim 41, wherein the method

further comprises marking, via the first table, the first fragment as invalid while the

creating and copying operations are being performed.

43. (Previously presented)   The machine-readable medium of claim 42, wherein the method

further comprises:

   determining whether the instruction has been cancelled before the creating and

      copying operations are completed;

   marking the first fragment as valid if the instruction has been cancelled before the

      creating and copying operations are completed; and

deleting the second fragment and the second table after marking the first fragment as valid.

44. (Previously presented)   The machine-readable medium of claim 41, wherein the method further comprises deleting the first table when the creating and copying operations are completed.

45. (Previously presented)   The machine-readable medium of claim 41, wherein the method further comprises marking, via the second table, the second fragment as a backup copy of the first segment while the creating and copying operations are being performed.

46. (Previously presented)   The machine-readable medium of claim 45, wherein the method further comprises unmarking the second fragment as the backup copy of the first segment after the creating and copying operations are completed.

47. (Previously presented)   The machine-readable medium of claim 41, wherein the copying the first table comprises writing the handle into a replacement entry.

48. (Previously presented)   The machine-readable medium of claim 47, wherein the writing the handle comprises marking a handle pointing to the first fragment as invalid and marking the handle pointing to the second fragment as valid.

49. (Previously presented)   The machine-readable medium of claim 41, wherein the first table is a member of a first chain having a plurality of tables, the method further

comprises: <u>A machine-readable medium having executable code to cause a machine to</u>

<u>perform a method, the method comprising:</u>

<u>receiving an instruction for updating a file having a plurality of fragments stored in a</u>

<u>memory, the instruction including changed data for updating a first fragment of</u>

<u>the file;</u>

<u>creating a second fragment based on the first fragment and the changed data;</u>

<u>copying a first table associated with the first fragment to form a second table having a</u>

<u>handle referencing the second fragment, the first table being a member of a</u>

<u>first chain having a plurality of tables;</u>

<u>deleting the first fragment when the creating and copying operations are completed;</u>

creating a copy for each of the plurality tables in the first chain to form a second chain,

each of the copied table having a handle pointing to the next copied table in the

second chain;

replacing the first chain with the second chain; and

deleting the first chain.

50. (Previously presented)   The machine-readable medium of claim 49, wherein each table

in the first and second chains is hierarchically ranked at a different level and wherein each

table is capable of associating with one or more tables in the respective chain via one or

more handles respectively.

51. (Currently Amended)   A data storage system, comprising:

a processor;

a memory coupled to the processor;

a ~~process~~ machine executable code, when executed by the processor from the memory,

~~having~~ causes a data object manager to

receive an instruction for updating a file having a plurality of fragments stored

in the memory, the instruction including changed data for updating a

first fragment of the file,

create a second fragment based on the first fragment and the changed data, the

second fragment representing an updated version of the first fragment,

copy a first table associated with the first fragment to form a second table, the

second table having a handle referencing the second fragment, ~~and~~

delete the first fragment when the creating and copying operations are

completed,

process further accesses to the deleted first fragment using the second fragment

via the second table.

52. (Previously presented)   The system of claim 51, wherein the data object manager further

marks, via the first table, the first fragment as invalid while the creating and copying

operations are being performed.

53. (Previously presented)   The system of claim 52, wherein the data object manager is

further to:

determine whether the instruction has been cancelled before the creating and copying

operations are completed;

mark the first fragment as valid if the instruction has been cancelled before the

creating and copying operations are completed; and

delete the second fragment and the second table after marking the first fragment as valid.

54. (Previously presented)   The system of claim 51, wherein the data object manager further deletes the first table when the creating and copying operations are completed.

55. (Previously presented)   The system of claim 51, wherein the data object manager further marks, via the second table, the second fragment as a backup copy of the first segment while the creating and copying operations are being performed.

56. (Previously presented)   The system of claim 55, wherein the data object manager further unmarks the second fragment as the backup copy of the first segment after the creating and copying operations are completed.

57. (Previously presented)   The system of claim 51, wherein the copying the first table comprises writing the handle into a replacement entry.

58. (Previously presented)   The system of claim 57, wherein the writing the handle comprises marking a handle pointing to the first fragment as invalid and marking the handle pointing to the second fragment as valid.

59. (Currently Amended)   ~~The system of claim 51, wherein the first table is a member of a first chain having a plurality of tables, wherein the data object manager is further to:~~ A data storage system, comprising:

a processor;

a memory coupled to the processor;

a machine executable code, when executed by the processor from the memory, causes

a data object manager to

receive an instruction for updating a file having a plurality of fragments stored

in a memory, the instruction including changed data for updating a first

fragment of the file,

create a second fragment based on the first fragment and the changed data,

copya first table associated with the first fragment to form a second table

having a handle referencing the second fragment, the first table being a

member of a first chain having a plurality of tables,

delete the first fragment when the creating and copying operations are

completed,

create a copy for each of the plurality tables in the first chain to form a second

chain, each of the copied table having a handle pointing to the next

copied table in the second chain[[;]],

replace the first chain with the second chain[[;]], and

delete the first chain.

60. (Previously presented)   The system of claim 59, wherein each table in the first and

second chains is hierarchically ranked at a different level and wherein each table is capable

of associating with one or more tables in the respective chain via one or more handles

respectively.